PATENT
APPLICATION 09/822,300
ATTORNEY DOCKET 2000P07515US01 (1009-087)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s)	:	Muenzel, Georg
Application #	:	09/822,300
Confirmation #	:	8037
Filed	:	23 March 2001
Application Title	:	Industrial Automation System Graphical Programming Language Storage and Transmission
Art Unit #	:	2193
Latest Examiner	:	Vu, Tuan A.

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. § 1.132

Sir:

I, Georg Muenzel, a citizen of Germany, whose full post office address is 75 Edgemere Ave, Plainsboro, NJ 08536, declare as follows under penalty of perjury.

Background

1. I hold a Dipl. (Univ.) [equivalent: Master's] degree in Mathematics from University of Erlangen-Nuernberg, Germany, awarded in 1982.
2. I am currently a Program Manager with Siemens Corporate Research, Princeton.
3. Since 1984, I have worked continually in the field of Industrial Automation, with particular emphasis in industrial automation engineering.
4. During my career, I have been granted one U.S. patents for my own inventions in the field of industrial automation systems.

Review

5. I have reviewed Application Serial No. 09/822,300 (hereinafter the present application).
6. I know what one of ordinary skill in the art of the present application would have known on the priority date claimed by the present application (24 March 2000).
7. I have reviewed the USPTO Office Action dated 14 August 2007 (hereinafter the "Office Action") regarding Application Serial No. 09/822,300.
8. I have reviewed U.S. Patent 6,634,008 ("Dole").
9. I have reviewed U.S. Patent 7,089,530 ("Darkinski").
10. Among the subject matter with which I was familiar prior to 24 March 2000 was subject matter of the type recited in Dole.
11. Among the subject matter with which I was familiar prior to 24 March 2000 was subject matter of the type recited in Darkinski.

Each of claims 36-38 is Definite

12. Claim 36, from which each of claims 37 and 38 ultimately depends, states, *inter alia*, "causing a programmable logic controller to control an industrial process via execution of an industrial automation computer program developed via a markup language version of the industrial automation computer program, the industrial automation computer program created using a graphical programming language".
13. The Office Action asserts, at Page 3 "[c]laim 36 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite in that it fails to point out what is included or excluded by the claim language. The reciting of 'computer program' having instructions for "causing a programmable logic controller **to control** an industrial process ...' (re claim 36) appears to be preemptive functionality that reads onto any similar endeavor by any prior art of related field, and this is not sufficiently setting forth the definiteness expected to reasonably construe what the invention is capable of in terms of the specific extent in functionality for this claimed computer product.

In other words, claim 36 is as being incomplete for omitting essential elements and/or cooperative relationships of elements that reasonably convey the realization of the 'to control' limitation, such omission amounting to a gap between the necessary structural connections. See MPEP § 2172.01. The omitted structural cooperative relationships are: the relationship between the controlling of an industrial process by a computer program operating on a PLC and the very use of markup language in developing the program. Further, the very broad terminology recited as 'to control' is considered not a specific limitation but rather a concept/limitation that covers a large range of actions. Indeed, a broad range or limitation together with a narrow range or limitation that falls within the broad range or limitation (in the same claim) is considered indefinite, since the resulting claim does not clearly set forth the metes and bounds of the patent protection desired".

14. The Office Action further asserts, regarding claim 36, at Page 4, "claim 36 recites the broad recitation 'program ... causing a programmable logic controller to control a industrial process', and the claim also recites 'computer program developed via a markup language version of the ... computer program' which is the narrower statement of the range/limitation. One of ordinary skill in the art would not be taught how a program developed via a markup version thereof can enable this very program to cause a industrial controller to control an industrial process, which appears to be a very broad endeavor in the absence of any more specifics. For example, it is indefinite as to how a markup construct can enable a program to cause a PLC to control an industrial process."
15. One skilled in the art would have found the assertions of the Office Action recited in each of paragraphs 13 and 14 factually incorrect as of 24 March 2000.
16. One skilled in the art would have reviewed at least FIG. 1, FIG. 2, paragraph 27, and paragraphs 36-42 of the published application (U.S. Patent Publication 20020004804) in order to understand the use and scope of the claimed subject matter, "causing a programmable logic controller to control an industrial process via execution of an

industrial automation computer program developed via a markup language version of the industrial automation computer program".

17. One skilled in the art would have reviewed at least paragraphs 3 and 4 of the published application (U.S. Patent Publication 20020004804) in order to understand the use and scope of the claimed subject matter, "the industrial automation computer program created using a graphical programming language".
18. One skilled in the art would have found that claim 36, as well as each of claims 37 and 38, when read in light of the specification at least at the locations indicated in each of paragraphs 16 and 17, reasonably apprises those skilled in the art of the use and scope of the claimed subject matter and that the language of each of claims 36-38 is as precise as the subject matter of each of claims 36-38 permits.

Each of Claims 51 and 52 is Definite

19. Claim 51, from which of claim 52 ultimately depends, states, *inter alia*, "a set of markup language tags associated with the markup language version of the industrial automation computer program defined for the graphical programming language, the set of markup language tags one of a plurality of sets of markup language tags, each set of markup language tags of the plurality of sets of markup language tags defined for a corresponding graphical language of a plurality of graphical languages used in industrial automation; and causing the programmable logic controller to control an industrial process via the industrial automation computer program".
20. The Office Action asserts, at Page 3 "[c]laim 51 (along with claim 52) is rejected because of the indefinites identified in the reciting of the newly added phrase '*a set of markup language tags associated with the markup language version of the industrial automation computer program defined for the graphical programming language, the set of markup language tags one of a plurality of sets of markup language tags, each set of markup language tags of the plurality of sets of markup language tags defined for a corresponding graphical language of a plurality of graphical languages used in*

industrial automation'. One of ordinary skill in the art would not be apprised on the invention metes and bounds in terms of relationship between the elements recited because of what appears to be an incongruous amalgamation of fragmented teachings (see Claim Objection), nor is one able to construe a minimal functionality concerning the plurality of tags recited within the onset of 'receiving' recited in the paragraph in order to make use of the plurality of tags presented as set each defined for a corresponding language of another plurality of languages. The above phraseology is treated as though the plurality of tags would each represent a graphical language of some type, among more language types or families".

21. One skilled in the art would have found the assertions of the Office Action recited in paragraph 20 factually incorrect as of 24 March 2000.
22. One skilled in the art would have reviewed at least paragraph 27, and paragraph 34 of the published application in order to understand the use and scope of the claimed subject matter, "a set of markup language tags associated with the markup language version of the industrial automation computer program defined for the graphical programming language, the set of markup language tags one of a plurality of sets of markup language tags, each set of markup language tags of the plurality of sets of markup language tags defined for a corresponding graphical language of a plurality of graphical languages used in industrial automation".
23. One skilled in the art would have reviewed at least FIG. 1, FIG. 2, and paragraphs 36-42 of the published application in order to understand the use and scope of the claimed subject matter, "causing the programmable logic controller to control an industrial process via the industrial automation computer program".
24. One skilled in the art would have found that claim 51, as well as claim 52, when read in light of the specification at least at the locations indicated in each of paragraphs 22 and 23, reasonably apprises those skilled in the art of the use and scope of the claimed subject matter and that the language of each of claims 51 and 52 is as precise as the subject matter of each of claims 51 and 52 permits.

Dole is Not Pertinent Art to the Claimed Subject Matter of Claims 1-52

25. The claimed subject matter of each of claims 1-52 relates to a method and system for standardized storage of graphical programming languages. *See* Page 1.
26. Dole is directed toward (emphasis added), "[a]n **environment for designing integrated circuits**. Computers include browsers for displaying pages of forms, with the computers in communication with a methodology server and a compute server. The methodology server contains design methodologies accessed by the computers, with the design methodologies defining steps of designing and testing of integrated circuits. The computers or methodology server are also in communication with a compute server. The compute server executes electronic design automation tools as requested." *See*, Abstract.
27. One skilled in the art would not have found, as of 24 March 2000, "[a]n **environment for designing integrated circuits**" to be reasonably pertinent to any problem related to the claimed subject matter, and in particular not to any disclosed problems associated with a method and system for standardized storage of graphical programming languages.

Dole in View of Dardinski and/or Allegedly Admitted Prior Art Do Not Render Claims 1-50 Obvious

28. Claim 1, from which each of claims 2-18 ultimately depends, recites, *inter alia*, "converting the internal representation to a markup language version of the industrial automation computer program" and "the industrial automation computer program adapted for controlling a programmable logic controller".
29. Claim 19, from which each of claims 20-35 ultimately depends, recites, *inter alia*, "an industrial automation computer program adapted for controlling a programmable logic controller" and "converting the identified industrial automation computer program from the internal representation to a markup language version of the

industrial automation computer program".

30. Claim 36, from which each of claims 37-39 ultimately depends, recites, *inter alia*, "causing a programmable logic controller to control an industrial process via execution of an industrial automation computer program developed via a markup language version of the industrial automation computer program".
31. Claim 39, from which claim 40 ultimately depends, recites, *inter alia*, "converting the industrial automation computer program, stored in memory in the internal representation, from the internal representation to a markup language version of the industrial automation computer program" and "the industrial automation computer program adapted for controlling a programmable logic controller".
32. Claim 41, from which each of claims 42 and 43 ultimately depends, recites, *inter alia*, "creating a schema defining a content model for a markup language version of an industrial automation computer program converted from a graphical language version of the industrial automation computer program, the industrial automation computer program adapted for controlling a programmable logic controller".
33. Claim 44, from which each of claims 45 and 50 ultimately depends, recites, *inter alia*, "accessing a markup language version of the industrial automation computer program, the markup language version of the industrial automation computer program converted from a representation created using a graphical programming language, the industrial automation computer program adapted for controlling a programmable logic controller".
34. Claim 51, from which claim 52 ultimately depends, recites, *inter alia*, "receiving data from the plurality of industrial automation program developer systems, the data comprising an industrial automation computer program presented in a markup language version, the markup language version of the industrial automation computer program converted from a representation created using a graphical programming language, the industrial automation computer program adapted for controlling a programmable logic controller".

35. The present Office Action asserts that the claimed subject matter of each of claims 1, 19, 36, 39, 41, 44, and 51 referenced in paragraphs 28-34 is taught by Dole at, col. 7, lines 26-42; Fig. 10; col. 16, lines 10-67; Fig. 13 and/or col. 16, line 65 – col. 17, line 2. See page 6, page 9, page 10, page 11, page 12, and page 13.
36. One skilled in the art would have found the assertions of the Office Action recited in paragraph 35 factually incorrect as of 24 March 2000.
37. One skilled in the art would have found that col. 7, lines 26-42 of Dole states, "[t]he interface and flow control tool encompasses HTML pages and CGI scripts. The HTML pages include input forms for defining methodologies, including steps of methodologies, as well as chip and block home pages and executable methodologies. The CGI scripts receive and act on data input to the input forms to create files defining methodologies, chips and blocks, and executable methodologies attached to chips and blocks. The CGI scripts also cause execution of electronic design automation (EDA) tools residing on the compute servers (illustrated in FIG. 2). Accordingly, the design server contains files 303. The files are created by the CGI scripts in response to input to the input forms applying new methodologies, and responsive to input to input forms attaching methodologies to chips or blocks. In addition, in one embodiment the files include files and libraries comprising design data formed as the result of the execution of the EDA tools."
38. One skilled in the art would not have found the applied portion of Dole cited in paragraph 37 to teach anything regarding a "markup language version" of an "industrial automation computer program" as claimed by each of claims 1, 19, 36, 39, 41, 44, and 51 as one skilled in the art would interpret the phrase "industrial automation computer program" that is required by each of claims 1, 19, 36, 39, 41, 44, and 51 to be "adapted for controlling a programmable logic controller" or adapted to control "an industrial process via" the "programmable logic controller".
39. One skilled in the art would have found that Fig. 10 of Dole illustrates:

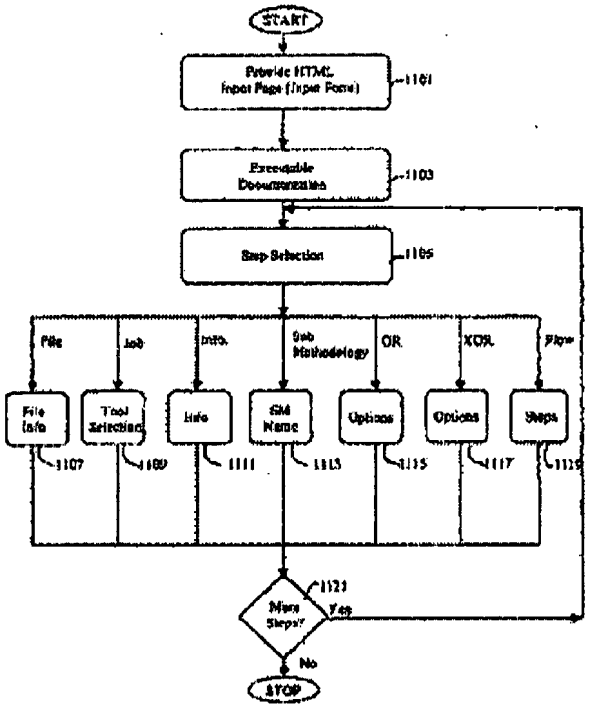

FIG. 10

40. One skilled in the art would not have found the applied portion of Dole cited in paragraph 39 to teach anything regarding a "markup language version" of an "industrial automation computer program" as claimed by each of claims 1, 19, 36, 39, 41, 44, and 51 as one skilled in the art would interpret the phrase "industrial automation computer program" that is required by each of claims 1, 19, 36, 39, 41, 44, and 51 to be "adapted for controlling a programmable logic controller" or adapted to control "an industrial process via" the "programmable logic controller".

41. One skilled in the art would have found that col. 16, lines 10-67 of Dole states, "[a]lthough HTML based forms are typically used to capture design methodologies, it is often more desirable to use XML (Extensible Markup Language) script to define design methodologies because of advantages that XML has over HTML. In XML, information is divided into useful components called elements, e.g., titles, paragraphs

and part numbers. The elements may be formatted, sorted, or searched in consistent fashion. The elements are typically named and defined in a computer program called a Document Type Definition (DTD) Using XML, a methodologist is able to create a single file to describe each design methodology. The single file that describes the design methodology may be used to create other files needed to execute the design methodology. For example, new features can be added to XML over time since XML is an extensible language. In addition, parsers are easy to develop using XML. For example, the parsers may be partially generated automatically from the DTD. Further, XML sources may be scanned by various different programs for different purposes. For example, a source code based on XML may be scanned by a search engine. Another benefit of using XML is that XML is capable of providing multiple language support. For another example, an XML file is easy to create provided that a good DTD has been created. In addition, an XML-based DTD file may be used to specify the internal nature of the XML files used to define design methodologies. Further, XML hyper-linking is more powerful than HTML hyper-linking, and XML hyper-linking may be used to refer to parts of other XML files. Widely used web browsers may not have a capability to display pages having embedded XML. Therefore, in an alternate embodiment, rather than using an input page to capture design methodology, a methodologist creates an XML script defining a design methodology in a single file. In this embodiment, the XML files are used by Common Gateway Interfaces (CGI's) to drive the integrated circuit design and fabrication system rather than directly viewed using a browser. FIG. 13 is a process of using XML as a design methodology capture script. According to the process in step 521, a methodologist captures a design methodology in a single file in the form of an XML script. Next, a converter with XML parsing capability is used in step 523 to convert the captured design methodology into multiple files including info and index files as well as a directed acyclic graph (DAG) file. As shown in step 525, the methodologist may update the design methodology by modifying the XML file. Thus

modified, the XML file may be converted again into multiple files. Use of a single XML file is preferable to using multiple files generated by HTML forms since the single XML is easier to archive and maintain than the multiple files. The process in step 527 may convert the modified design methodology in XML format to the design methodology format where the design methodology comprises multiple files. In an embodiment of the present invention, a converter converts the captured design methodology format having multiple files to a single file having an XML script."

42. One skilled in the art would not have found the applied portion of Dole cited in paragraph 37 to teach anything regarding a "markup language version" of an "industrial automation computer program" as claimed by each of claims 1, 19, 36, 39, 41, 44, and 51 as one skilled in the art would interpret the phrase "industrial automation computer program" that is required by each of claims 1, 19, 36, 39, 41, 44, and 51 to be "adapted for controlling a programmable logic controller" or adapted to control "an industrial process via" the "programmable logic controller".
43. One skilled in the art would have found that Fig. 13 of Dole illustrates:

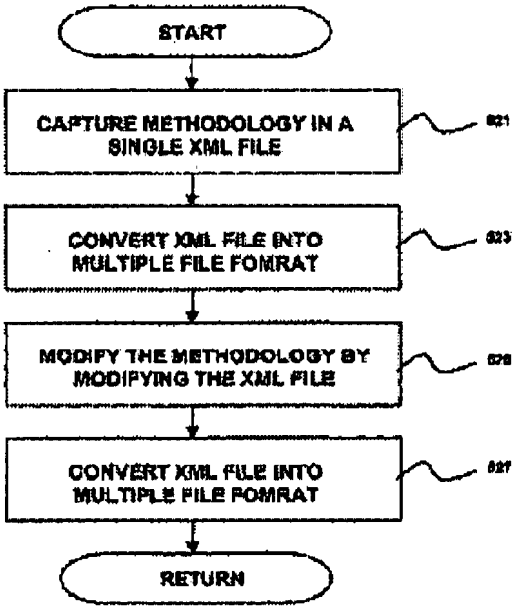

FIG. 13

44. One skilled in the art would not have found the applied portion of Dole cited in paragraph 37 to teach anything regarding a "markup language version" of an "industrial automation computer program" as claimed by each of claims 1, 19, 36, 39, 41, 44, and 51 as one skilled in the art would interpret the phrase "industrial automation computer program" that is required by each of claims 1, 19, 36, 39, 41, 44, and 51 to be "adapted for controlling a programmable logic controller" or adapted to control "an industrial process via" the "programmable logic controller".
45. One skilled in the art would have found that col. 16, line 65 – col. 17, line 2 of Dole states, "[i]n an embodiment of the present invention, a converter converts the captured design methodology format having multiple files to a single file having an XML script. A specification of a Methodology Document Type Definition (DTD) is used during development of these converters."
46. One skilled in the art would not have found the applied portion of Dole cited in

paragraph 37 to teach anything regarding a "markup language version" of an "industrial automation computer program" as claimed by each of claims 1, 19, 36, 39, 41, 44, and 51 as one skilled in the art would interpret the phrase "industrial automation computer program" that is required by each of claims 1, 19, 36, 39, 41, 44, and 51 to be "adapted for controlling a programmable logic controller" or adapted to control "an industrial process via" the "programmable logic controller" as one skilled in the art would interpret the phrase "industrial automation computer program" that is required by each of claims 1, 19, 36, 39, 41, 44, and 51 to be "adapted for controlling a programmable logic controller" or adapted to control "an industrial process via" the "programmable logic controller".

I further declare that all statements made herein of my own knowledge are true and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code and that willful false statements may jeopardize the validity of the application or any patent issuing thereon.

Signed this 28th day of October, 2007

Georg Muenzel